https://www.financial-planning.com/news/interested-in-vibe-coding-heres-how-advisors-can-start

Interested in vibe coding? Here's how advisors can start
By Rob Burgess
Published March 05, 2026, 12:41 p.m. EST

Coding is a term that conjures thoughts of significant technical know-how. But vibe coding offers advisors a way to get the same output through natural language.

Though it may sound complicated, advisors with experience in vibe coding say it's doable for most of their peers, especially with free or relatively cheap tools available. And it's allowed many to fill in the gaps of their technology stacks without major software overhauls.

Selecting the right tools
The most widely used and stress-tested tools for vibe coding are either ones that operate directly using a custom editor — such as Cursor, Windsurf or Antigravity — or those that have their own chat interface or tooling baked into an existing interface — such as Codex or Anthropic's Claude Code — said Archie Chaudhury, CEO and co-founder of AI evaluation firm LayerLens.

"It certainly seems that the latter has overtaken the former both from a user experience and functionality standpoint," he said. "Increasingly, we will see such tools that are able to take actions on a user's machine or workstation directly become more and more popular."

Tim Witham, founder of Balanced Life Planning in Villa Hills, Kentucky, said he uses Claude Max, which is $100 per month for vibe coding, and has settings for Chat, Code and Cowork. (Code is sophisticated and the most powerful. Chat is easy to use and can produce things, but not execute them. Cowork is a blend between the two, he said.)

Naz Avo, founder of employee engagement platform FeedbackPulse, built the product using AI-assisted tools for almost all of the code. In addition to Claude, he also suggested using a development environment like Visual Studio Code if they care to get more technical. (Claude comes with native development environment integration and is useful in Cowork and Code, he said.)

"This single suite of tooling is more than enough to get one going," he said.

Steven Crane, founder of Financial Legacy Builders in Dayton, Ohio, said he is using vibe coding to build internal financial analysis tools. He said he primarily uses AI coding tools like ChatGPT and occasionally Microsoft Copilot to help him build a customized financial analyst in which he feeds structured client data into a secure environment and generates clean financial snapshots, income summaries, spending breakdowns, net worth statements, debt schedules and

cash flow maps.

Shaun Melby, founder of Melby Wealth in Nashville, Tennessee, said if advisors are curious about it but don't know where to start, just start. He said he recommended signing up for the free versions of Replit and Loveable and start telling them what to build.

"From my experience, the best way to learn is to start trying it yourself," he said. "You'll make mistakes, but you'll learn from them."

Making a plan

Vibe coding is a journey, and users are probably not going to come up with the perfect prompt and produce the desired result in one go, said Witham.

To get started, he said, it's best to feed AI with who you are, what you're looking to accomplish, an example and an indication of whether it should be a static or dynamic (fixed or ever-evolving) result, he said.

Beginning is a matter of collecting these thoughts and data points and putting in the first prompt, said Witham.

"The AI will come back with something," he said. "It's important to take some time and review it. It can and will make mistakes and will require edits."

Crane said his vibe coding process is straightforward:

1. Define what you want the output to look like.
2. Outline the logic in plain English.
3. Have the AI generate the code for a simple tool, often in Python or basic web format, that organizes and summarizes the data.
4. Test it with sample data before ever using real numbers.
5. Refine it so the output is clean and useful.

The refining process

Where things tend to break down is when users skip the intent-setting step entirely and let the tool make every decision, said Chaudhury.

"[This] is how you end up with applications that work on the surface but lack basic considerations like authentication, error handling or scalability," he said.

Vibe coding is not about removing the need to think through what you are building, said Chaudhury.

"It is about removing the gap between knowing what you want and being able to ship it," he said.

Once users receive the first response from the AI, Witham said it's important to make a list of elements that need to change. Once users enter them, he said they

should then evaluate the next output.

"If it generates something you're not sure about, ask it to give you the methodology used," he said. "Your experience and knowledge is needed to guide AI to the solution — you can't be mesmerized by its power."

Once users have a working prototype, Chaudhury said the testing process is less about running a formal quality assurance suite and more about treating yourself as the end user and stress-testing the tool against real scenarios.

"You start by interacting with it the way someone actually would," he said. "Try the core workflows, push edge cases, enter unexpected inputs and see where it breaks. And it will break."

When it is completed, Crane said he stress tests everything. This includes checking totals manually and comparing outputs against trusted spreadsheets.

Crane said he even deliberately inputs strange data to see if it breaks. If something looks off, he then goes line by line and fixes the logic, he said.

"No client ever sees something that has not been verified by hand," he said. "The goal is not to automate advice. The goal is to automate the organization. That way, I can focus on what actually moves the needle, which is coaching, accountability and helping people fight their way out of whatever financial hole they are in."

The reality of vibe-coded tools is that the first version almost always has gaps, whether that is a form that fails silently, an application programming interface (API) call that does not handle errors gracefully or a user interface (UI) element that behaves differently than described, said Chaudhury.

"When you hit an error, the fix follows the same iterative loop as the build itself," he said. "You describe the problem to the coding tool, provide context on what the expected behavior should be and let it generate a fix."

For straightforward bugs, this works well, but for more structural issues, like flawed data handling or a misunderstood integration, users may need to step back and redescribe the architecture or the specific requirement more precisely, said Chaudhury.

"The key discipline here is to not just accept the first fix blindly," he said. "Vibe coding does not eliminate debugging. It changes the medium."